WizePanel™
Adapter

(rev. 405)

Documentname: WizePanel_Adapter_English
(revision: 405, filesize: 57 kB)
(created @ 26/04/2012 07:01:00 by Artur Wiebe)
(last modified and saved @ 2012-05-25 13:32 by Artur Wiebe)
WTDN:

Wilke Technology GmbH	*Phone:*	*+49-2405 / 55-0*
Heider-Hof-Weg 23D	*Fax:*	*+49-2405 / 55-44*
52080 Aachen	*Mail:*	*info@wilke.de*
Germany	*Web:*	*www.wilke.de*

Content

1 Introduction

The purpose of the WizePanel adapters is to retrieve external data sources, process the data and convert it to images. The images are finally displayed on the WizePanels. The following graph gives a basic overview of the process.

```
                    WizePanel-Server
  ┌──────────┐   ┌──────────┐   ┌──────────┐   ┌──────────┐
  │ External │ ← │ Adapter  │   │  Image   │ → │ WizePanel│
  │Data Source│   └────┬─────┘   └────▲─────┘   └──────────┘
  └──────────┘        │              │
                 ┌────▼─────┐  +  ┌───┴──────┐
                 │  Event   │  →  │ Template │
                 └──────────┘     └──────────┘
```
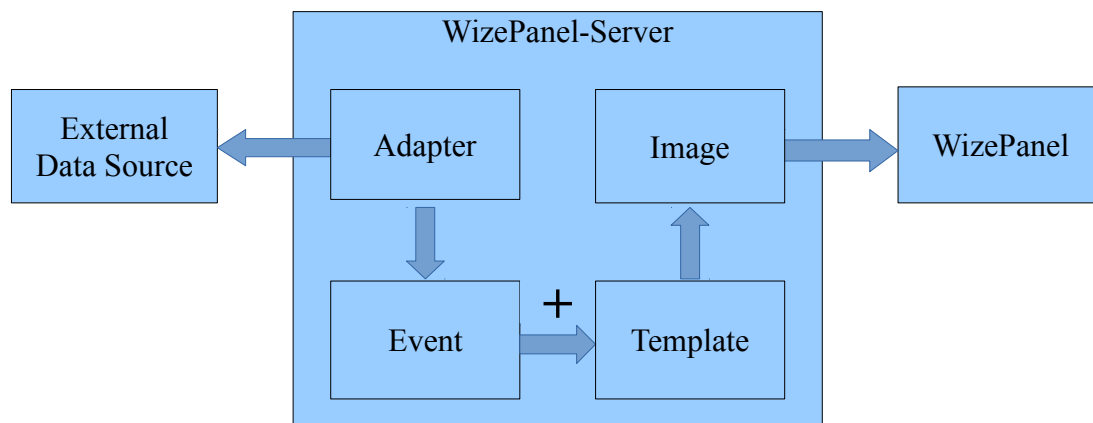
Different adapters are available for different data sources. The adapter monitors an external data source, which can be anything starting with a file, directory or a HTTP server.

The data retrieved is converted into a calendar-like event which contains essential information like the start and end of the event. Additionally events typically have a title, subject or a summary. Depending on the adapter type many other information may exist.

Once the event is added to the system, at the time the event takes place, data of this event and others, is used to fill a template. Templates describe the layout of events on the screen.
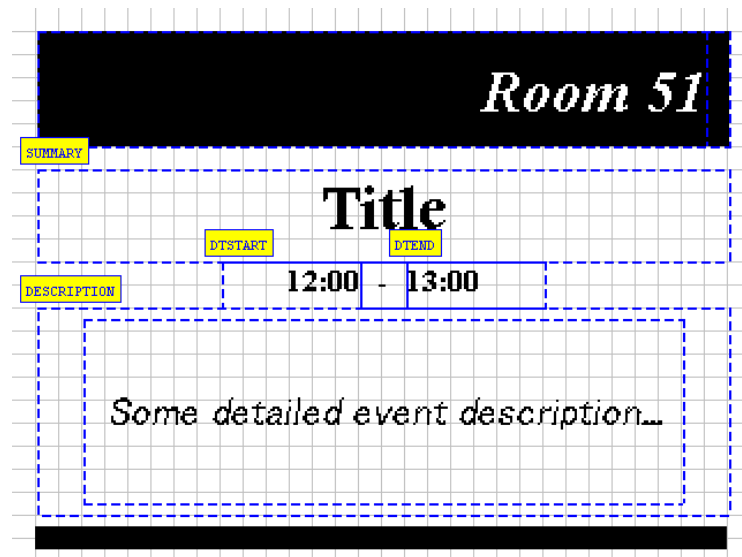
Take a look at the this template.

Blue, dotted, boxes surround text fields. Some of the text fields have names which you can see in yellow boxes: SUMMARY, DTSTART, DTEND and DESCRIPTION. These are called place-holders. Their content will be set when the final image for a WizePanel is created. "Room 51" and "-" text fields do not have a name – their content cannot be changed.

Lets say that you have an event that starts at 16:00 and ends at 17:00. It's a meeting concerning a meeting about WizePanels. This is how the generated image will look like.

1.1 WizePanels

WizePanels are referred to by their name as displayed in the Server-Explorer in the Wize-Panel-Studio.

1.2 Events

Basically events are described by a start and an end and have a unique ID. The ID is requires to identify changes made to the event. Anything else is optional but typically events have a title (subject, summary). It depends on the adapter which data is available. Some are fixed, others are free to be defined.

A number of adapters support private events. In this case these adapters allow to specify an alternative title on the WizePanel.

1.3 Templates

Templates describe how the final images will be constructed using placeholders which are filled using information from events.

It is possible to display more than one event in a template by specifying it in square parenthesis [x] where it starts with 0 for the current event and 1 for the next event.

This is a set of default placeholders:

DTSTART[0..n]	Start of the event
DTEND[0..n]	End of the event
SUMMARY[0..n]	Summary, subject, title
DESCRIPTION[0..n]	Detailed description, body
LOCATION[0..n]	Location
CREATOR[0..n]	Creator
DATETIME[0..n]	Contains start and stop time. Therefore this placeholder must contain two date formatters. Default: {date:HH:mm} − {date:HH:mm}
EVENT_QR[0..n]	Calendar appointment of the this event
TARGET_NAME	Name of target
FREE_UNTIL	If there is no current event this time is set to the next upcoming event time.

TODAY	Current day
TARGET_COMMENT	The whole comment. Can easily be split into single lines by {content:line1} etc...
SOURCE	The name of the source (without the adapter name).

Custom placeholders are created in the same way.

It is possible to format some of the placeholders. The format string looks always like this: "text {format pattern} text".

Here an incomplete list of typical usage:

Format Pattern	Output
date:HH:mm	12:50
date:dd.MM.yyyy	29.06.2013
date:dd.MM.	29.06.
date:dd.MM.yyyy HH:mm	29.06.2013 12:50
date:d. MMMMM	29. May
date:d. MMMMM yyyy	29. May 2013
date:EEE.	Sun.
content:none	Content
content:upper	Upper case version of the content
content:lower	Lower case version of the content
content:line1 content:line2 content:lineX	To split the content by lines

See http://docs.oracle.com/javase/7/docs/api/java/text/SimpleDateFormat.html for full list of patterns.

1.4   Multiple Sources

It is possible to specify more than one data source for a WizePanel. Doing this changes the way placeholders are named. Each placeholder is prefixed with the ID as seen in the target configuration, followed by a "#": **ID#PLACEHOLDER[0..n]**.

Example: There are two sources and each has three events. The following four SUMMARY placeholders will be generated.

Source 1: 0#SUMMARY[0] and 0#SUMMARY[1] and 0#SUMMARY[2]

Source 2: 1#SUMMARY[0] and 1#SUMMARY[1] and 1#SUMMARY[2]

1.5   Licenses

Many adapters require licenses which allow the specified WizePanels to be processed by this adapter. Sometimes you must specify the WizePanels explicitly, e.g. CSV adapter or XML adapter. Then again, your adapter settings explicitly determine how many licenses you need, e.g. MS Exchange EWS adapter.

2   Adapters

2.1   MS Exchange 2003 – Auto-Accept-Agent [Events]

MS Exchange Server 2003 requires an installation of the additional add-on called Auto-Accept-Agent (get it for free from the Microsoft homepage). The purpose of this add-on is to accept any event reservations made on an accounts calendar. It saves any incoming events as emails in a folder. Our adapter retrieves these emails and creates events from them.

Once installed, the add-on has to be registered on every room account.

This adapter supports all default placeholders.

2.1.1   Installation

The scripts used below are included in the installation directory of the AutoAcceptAgent.

The steps required are as follows:

1. Install AutoAcceptAgent (in EXCHANGE/Agents/AutoAccept)

2. (Optional) Change configuration file AutoAccept.config.xml

3. Register your mailboxes by replacing MAILBOX@DOMAIN with your mailbox

    ```
    cscript RegisterMailbox.vbs /m:MAILBOX@DOMAIN
    ```

4. Use this command to show all registered mailboxes on the server named with EXCHANGE_SERVER:

    ```
    cscript ListMailboxes.vbs /S:EXCHANGE_SERVER
    ```

2.1.2 Configuration

You need the IP of the server and the mailbox address. Private events are also supported.

2.2 MS Exchange 2007 SP2+ – Exchange Web Services (EWS) [Events]

Starting with MS Exchange Server 2007 SP2 there is the Exchange Web Services interface (EWS) which is using SOAP messages. Using EWS it is possible to read calendars directly. The base address for the services looks like this:

```
http[s]://SERVER-ADDRESS/EWS/Exchange.asmx
```

"SERVER-ADDRESS" is to be replaced by the actual value.

Access through the HTTPS is possible but requires that Java run-time environment trusts used certificate. Self-signed certificates must be imported using the procedure is described in the section "Self-Signed Certificates".

This adapter supports all default placeholders.

2.2.1 Configuration

The configuration requires the server URL, a user and a password.

The adapter allows you to watch several calendars. Calendar must be the full folder name like meetingroom@myhost.com.

A special option "Range in days" describes how many days to look in advance. For instance, if you want to display events on the current day only, than set this option to 1. The days always end at midnight.

Furthermore, this adapter supports private events and the default placeholders.

Use the "Test" button to check the connection.

> **It is important that the user trying to access the calendar must have "Full Control" rights for it. And the same user is running the WP-Server and configured in the Exchange configuration to access the calendars.**
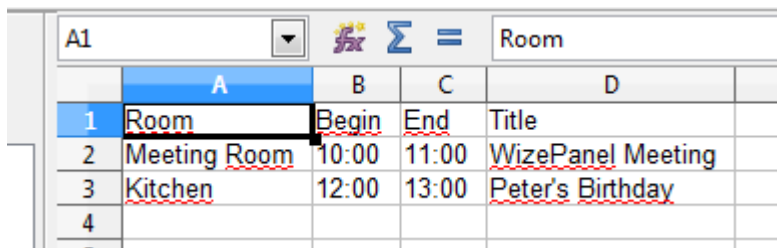
2.3 Excel/CSV [Events]

The CSV format is a simple text-based format for spread sheet programs. It has no visual gimmicks but provides data in lines and columns.

Each line represents a single event. Its columns are event fields, each describing one event property. Columns are separated by a special character, often a comma or semi-colon. This is how such a file looks like opened in a text editor.

```
Room;Begin;End;Title
Meeting Room;10:00;11:00;WizePanel Meeting
Kitchen;12:00;13:00;Peter's Birthday
```

The first line simply describes the type of the columns and will be skipped. Following are two events in two different locations. The first events takes place in the "Meeting Room", starting at 10:00 and ending at 11:00. Its title is "WizePanel Meeting". The second event is called "Peter's Birthday" and takes place in the kitchen from 12:00 to 13:00.

The same file opened in a spread sheep application:

The adapter is invoked when the file is changed only. That means that you need to change the file everyday if you want your events for that day.

Three placeholders are available by default: DTSTART, DTEND, TARGET_NAME. All other placeholders must be defined manually.

2.3.1 Configuration

Select the file you want to monitor.

Next, we have to define the format of the CSV file. Play with the parameters until you see you data in an ordinary table.

- SkipLines: Top lines are often used to describe the table and contain no events. These should be skipped.

- Separator: This char separates column. In most cases this is either the comma (,) or the semi-colon (;).

- Enclosure: Sometimes the column data is encoded in some sort of quotes ("). It won't hurt if this is not the case.

- Encoding: Encoding of the CSV file. Try this option if you do not see special characters of your language.

Now, you should see you data represented as a spread sheet. In the next step you must give columns meaning. Right-click on each column and select the type. The following screenshot displays the minimum configuration for the example CSV file. Only the column "Title" is missing. You can give it any name by selecting "…". The same placeholder must be available in the used template.
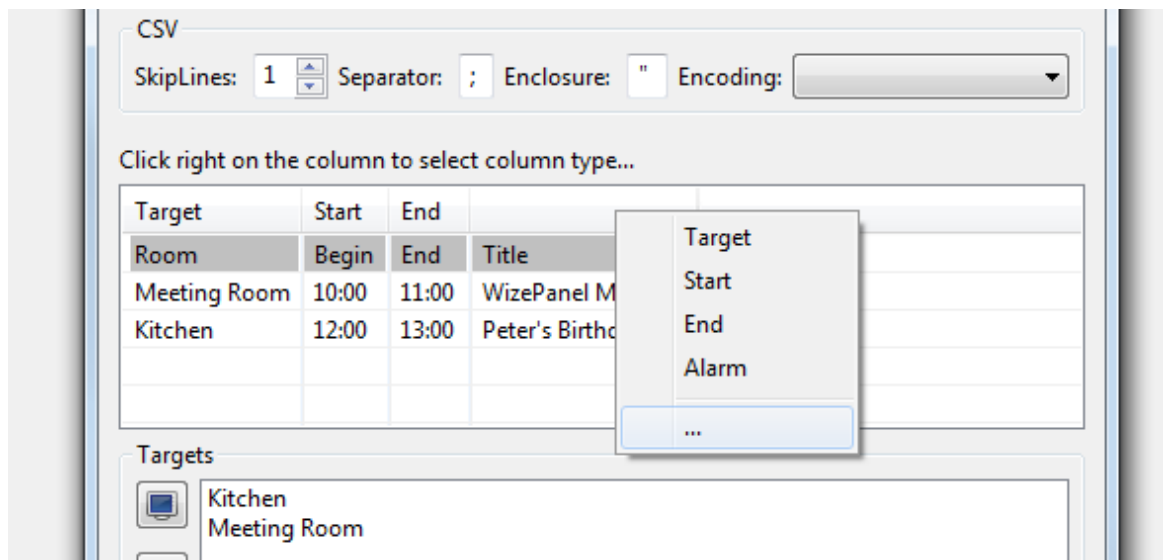
- Target: This is the column containing your WizePanel names.

- Start: Time when the event start. This can be time-only which will make the event to start on the current day. Times can be specified as "HH:mm" or in any format described in 5.1 Known Time Formats.

- End: Time when the event ends. This can also be time only to end the event on the current day. In case this time if later then "Start" the next day is the one to end it on. Times can be specified as "HH:mm" or in any format described in 5.1 Known Time Formats.

- Alarm: Time in minutes. Display the event some minutes earlier than it actually starts.

- Date: You can set the date in an extra column. Optional.

The license model requires that all WizePanels which will be served by this adapter must be defined in the targets list.

2.4 XML [Events]

The XML adapter monitors specified directory and any XML files in it. To remove an event remove the file.

This is how an XML file should look like::

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<WpEvent>
     <Start>2013-07-16 14:00:00</Start>
     <End>2013-07-16 15:00:00</End>
     <Alarm>5</Alarm>
     <SourceName>Room 51</SourceName>
     <Placeholders>
          <Placeholder>
               <Name>SUMMARY</Name>
               <Value>WizePanel Training</Value>
```

```
            </Placeholder>
            <Placeholder>
                    <Name>DESCRIPTION</Name>
                    <Value>WizePanelStudio 1.0</Value>
            </Placeholder>
        </Placeholders>
</WpEvent>
```

Fields explained:

- Start: Time to start.

- End: Time to end.

- Alarm: Optional. In order to start showing the event earlier specify the time in minutes here.

- SourceName: Name of the WizePanel as defined in the adapter.

- Placeholders: List of placeholders. All placeholders will be available in the template. DTSTART and DTEND cannot be used as they will be created from "Start" and "End" fields already.

Supported placeholders are DTSTART and DTEND. Others as defined.

2.4.1 Configuration

Specify the directory to monitor.

2.5 iCalendar [Events]

The iCalendar adapter monitors specified directory and any iCalendar (Extension: .ics) files in it. The source as defined in the WizePanel-Studio is the LOCATION field .

2.5.1 Configuration

Specify the directory to monitor.

2.6 DEA-EMS [Events]

Dean Evans & Associates event management systems (DEA-EMS), http://dea.com, is accessed using their SOAP interface. Tested with API Version 1.1.15.

Default placeholders, besides DESCRIPTION, are supported.

2.6.1 Configuration

Configuration requires the server address, user and password. Then the connection with the server is established and the list of available rooms is retrieved. Rooms with WizePanels must be selected. For each selected room a WizePanel with the appropriate name must exist. The address looks like this:

```
http://SERVER/service.asmx
```

If you are unsure, try the address in a web browser. It should display some weird tree-like output.

Maximum number of rooms is limited by the number of licenses.

This adapter allows you to specify an alternative event title for private events.

2.7 aSc TimeTables [Events]

aSc TimeTables is a school scheduling software (http://www.asctimetables.com/).

2.7.1 Configuration

Specify the address of the dailyplan. It should look like this: https://SERVER/connect_daily-plan.php?cmd=getdailyplan&date=2012-07-10

Default placeholders, besides DESCRIPTION, are supported.

2.8 HTTP-General (Wilke) [Content]

This is a very general adapter which reads data from a HTTP server using GET. After reading data it is removed from the database (or marked as read) and not returned on the next read attempt.

The response is a JSON list of JSON objects. "target_name" is the name of the WizePanel. Placeholders is a key-value list, actually JSON object.

```
[
    {
        "target_name": "name of the target",
        "placeholders": {
            "SUMMARY": "Title",
            "DESCRIPTION": "This and that…"
        }
    },
    {
        "target_name": "...",
        ...
    }
]
```

2.8.1 Configuration

Full address from which to get events.

2.9 Google Calendar [Events]

In order to run the Google Calendar adapter you need its **private ICS address**. You can find it in your calendar settings.

```
Important: Each time the whole calendar it retrieved, including expired
events. Therefore, for perfomance reasons, it is recommended to remove
these from time to time.
```

2.10 iCalendar [Events]

This adapter monitors a directory for ICS files. The LOCATION field is used as data source.

3   UPI - Low-Level-Company-Interface

Additionally, there is another low-level software interface which deals with Dispatchers and WizePanels bypassing the need for the WizePanel-Studio. This is a REST based service. It allows you to get a list of all Dispatchers and all the WizePanels they see. You can set the routing table, meaning which Dispatcher is responsible for which WizePanel. Finally, you can send your images to the WizePanels.

As this interface does not need the Studio component, you software will provide an appropriate GUI to manage the hardware.

You have to sign the WizePanel Trusted Software Partner Agreement in order to get a detailed description of the interface.

4   Self-Signed Certificates

Self-signed certificates must imported into the Java key store.

```
Important: You will have to repeat this step after each Java update.
```

4.1 Symptoms

If Java cannot verify a certificate it display this warning:

```
javax.net.ssl.SSLHandshakeException:
sun.security.validator.ValidatorException: PKIX path building failed:
sun.security.provider.certpath.SunCertPathBuilderException: unable to
find valid certification path to requested target
at com.sun.mail.imap.IMAPStore.protocolConnect(IMAPStore.java:441)
at javax.mail.Service.connect(Service.java:233)
at javax.mail.Service.connect(Service.java:134)
....
```

4.2   Get the Certificate

Now, you need to export the certicate into a file using a web browser. (Firefox works like a charm here. IE fails.) Navigate to the server in your browser and let it display the certificate. Look for a way to export it into X.509 format(PEM). I suggest to put the file temporarily in c:\, e.g. c:\my.pem. You can open the file with a text editor. It must look like this:

```
-----BEGIN CERTIFICATE-----
MIICiTCCAfKgAwIBAgIBADANBgkqhkiG9w0BAQQFADB/MQswCQYDVQQGEwJBVTEM
MAoGA1UECBMDTlNXMQ8wDQYDVQQHEwZTeWRuZXkxEjAQBgNVBAoTCUF0bGFzc2lh
bjEaMBgGA1UEAxMRY3ZzLmF0bGFzc2lhbi5jb20xITAfBgkqhkiG9w0BCQEWEmlu
Zm9AYXRsYXNzaWFuLmNvbTAeFw0wNTA5MjMwNjUyNTNaFw0wNjA5MjMwNjUyNTNa
MH8xCzAJBgNVBAYTAkFVMQwwCgYDVQQIEwNOU1cxDzANBgNVBAcTBlN5ZG5leTES
MBAGA1UEChMJQXRsYXNzaWFuMRowGAYDVQQDExFjdnMuYXRsYXNzaWFuLmNvbTEh
MB8GCSqGSIb3DQEJARYSaW5mb0BhdGxhc3NpYW4uY29tMIGfMA0GCSqGSIb3DQEB
AQUAA4GNADCBiQKBgQDhwAgx/gDgKe9tBjUCj7JtVkwQSzj2Dq0PHiJu1AWUYWFW
ivbBWaWSYbt/w9vIRSL8OlGVOLnlFOH5o7QIpIBZvd3xBMv6DxMijM86/hu8QTPt
KcMuqBTGpu1T846SzNncj883wSE1hXxezCgEFCsqyC7dVX4lOAy6zgzkt2wc3QID
AQABoxUwEzARBglghkgBhvhCAQEEBAMCBkAwDQYJKoZIhvcNAQEEBQADgYEAJOgg
O4brCcQa3IgONo8UmLcHo6Rq+Py6ZA3ueUegy/uyQ358JUeL4kktXuYL9gAPCuMc
hsC1iyaOrwY/S9S67w2Zwqc+uYX9ophFHkxK1r3YiaiMpEzMyB12VWSrOITcROLV
7NTWfxfPLUpkDbj+Mw/66QJkIOlqBvcKn3KXI74=
-----END CERTIFICATE-----
```

4.3   Import into Java

Start the windows console tool cmd.exe **as administrator** – search for the tool in your Windows Start, right-click on it and select "Run as Administrator...".

Go to your Java installation directory. Your directories may be different.

```
cd c:\
cd Program Files
cd Java
cd jre7
cd lib
cd security
```

Once you arrived you need to run the Java keytool command. **Important: It will ask you for a password which is "changeit" by default.**

```
..\..\bin\keytool -import -keystore cacerts -file c:\my.pem -alias my-
domain
```

This command assumes that your certificate is saved in "c:\my.pem". Alias is a memorable short-cut for the certificate. It can be used to remove certificates from the key store.

Now restart your applications.

5   Misc

5.1   Known Time Formats

Format	Example	Comment
yyyy-MM-dd HH:mm:ss Z	2013-06-29 12:50:00 CEST	
dd.MM.yyyy HH:mm	29.06.2013 12:50	
yyyy-MM-dd HH:mm:ss	2013-06-29 12:50:00	
yyyy-MM-dd HH:mm	2013-06-29 12:50	
yyyyMMdd'T'HHmmss'Z'	20130629T125000Z	Zulu Time
yyyyMMdd'T'HHmmss	20130629T125000	
yyyy-MM-dd'T'HH:mm:ss	2013-06-29T12:50:00	

Current time zone is used, if the time zone is not specified.

6   History

- Artur Wiebe, 05.08.2013 – Initial Version